# On Recommendation of Learning Objects using Felder-Silverman Learning Style Model

Shaimaa M. Nafea, François Siewe, Ying He

**Abstract**—The e-learning recommender system in learning institutions is increasingly becoming the preferred mode of delivery, as it enables learning anytime, anywhere. However, delivering personalised course learning objects based on learner preferences is still a challenge. Current mainstream recommendation algorithms, such as the Collaborative Filtering (CF) and Content-Based Filtering (CBF), deal with only two types of entities, namely users and items with their ratings. However, these methods do not pay attention to student preferences, such as learning styles, which are especially important for the accuracy of course learning objects prediction or recommendation. Moreover, several recommendation techniques experience cold-start and rating sparsity problems. To address the challenge of improving the quality of recommender systems, in this paper a novel recommender algorithm for machine learning is proposed, which combines students actual rating with their learning styles to recommend Top-N course learning objects (LOs). Various recommendation techniques are considered in an experimental study investigating the best technique to use in predicting student ratings for e-learning recommender systems. We use the Felder-Silverman Learning Styles Model (FSLSM) to represent both the student learning styles and the learning object profiles. The predicted rating has been compared with the actual student rating. This approach has been experimented on 80 students for an online course created in the MOODLE Learning Management System, while the evaluation of the experiments has been performed with the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The results of the experiment verify that the proposed approach provides a higher prediction rating and significantly increases the accuracy of the recommendation.

**Index Terms**—Recommendation system, collaborative filtering, content-based filtering, hybrid filtering, e-learning, rating prediction, felder-silverman learning style model.

## 1 Introduction

E-Learning Recommender Systems (E-LRS) have become popular in recent years. Compared with Learning Management Systems (LMS), which offer limited adaptivity and personalization, adaptive educational systems use intelligence algorithms to adapt to students learning style, enhance learning performance, accelerate goal achievement, reduce navigational overheads, and to enhance overall student satisfaction [1]. In addition, the adaptability and diversity of recommendations are desirable in e-learning recommender systems, because learners preferences and abilities keep changing. The diverse and adaptive Learning Objects (LOs) recommendations should be presented to motivate the learning potential of learners and to ensure a long-term learning experience [2], [3], [4].

A lot of learners are moving away from utilising e-learning systems, because they do not find them beneficial [26], [27], [28], [29].On particular, this is owing to the fact that this type of learning environment cannot interact with learners as well as the traditional face to face one. Moreover, learners usually make a set of choices during learning, for instance, ’how to learn’, ’with whom to learn’ and ’which learning pathway to follow’, and so on. To achieve this goal, it is essential to consider the students learning styles and use them in the design and implementation of e-learning environments, with the aim of making them more realistic and thus, attractive [30]. In this paper, we proposed a novel algorithm to recommend the most suitable course LOs taking into consideration student learning styles and LO profile. This paper is an extension of work originally presented in [70].

A recommender system (RS) enables users to cope with information overload by providing the most appropriate items based on their requirements. Figure 1 shows the traditional method of a two-dimensional recommender system, which has three main components: user, item, and rating. Rating, in this case, refers to the feedback that a user gives for a specific item, being implicit or explicit.

- *Explicit ratings* are when the user rates an item to express his/her level of interest. Rating can be in the form of a numeric value on a multi-point scale, e.g. 1 to 5 [83].
- *Implicit ratings* are generated by the RS itself, through inferences from users’ behaviour.

A user-item matrix is shown in Fig.1, where the elements in the matrix are the users ratings. In the matrix, the rows depict the user list, while the columns represent the items list. The numerical values from 1 to 5 in the matrix reflect the level of preference for a particular user for each item. The objective of RS algorithm in this setting is to predict the missing values in the matrix where users have not provided their preferences for certain items. However, this RS always suffers from data sparsity and cold start.

- *Shaimaa M. Nafea with the Department of management information system, Arab Academy For Science Technology and Maritime, Egypt. François Siewe and Ying He with the Department of Computer Science and Informatics, De Montfort University, Leicester, UK. E-mail: P15017421@myemail.dmu.ac.uk, fsiewe@dmu.ac.uk, ying.he@dmu.ac.uk*
-

*Manuscript received April 19, 2005; revised August 26, 2015.*

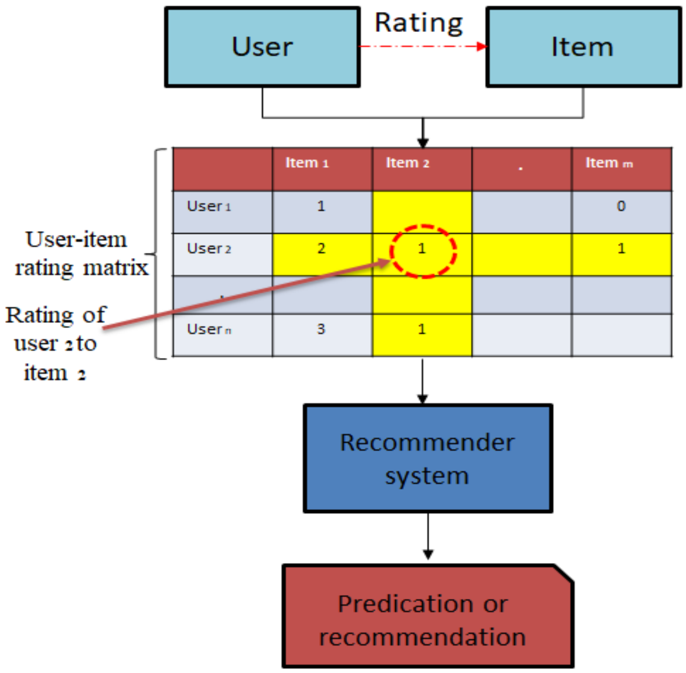



Fig. 1: A Traditional Recommendation Approach

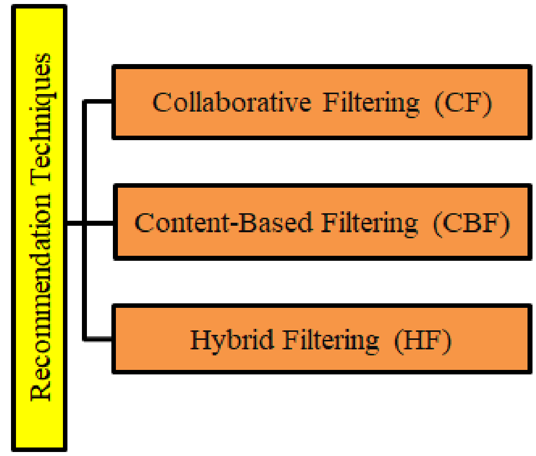



Fig. 2: Recommendation techniques

Data sparsity refers to the situation where the amount of information (ratings) of a target user is not sufficient enough to generate reliable related users (i.e. the number of commonly rated items among users is very small). While Cold start refers to the situation where an RS encounters new users or items with no ratings [100].

RSs have been researched and deployed extensively over the last decade in various application areas, including e-commerce [85], books [49], and movies [41], [48]. Another area where this support is very much demanded, is in the e-Learning field, where it is desirable for learners to be offered the most appropriate activities and learning objects to achieve their individual learning goals, whilst supporting their needs in the most efficient way. These kinds of RSs are usually focused on alleviating the information overload (of LOs) by filtering the most relevant content (LOs) to match the students preferences [2], [3], [4]. In sum, the major task of an RS is to construct a suitable model to calculate students' interests. RSs are widely classified into three main techniques in the literature, as shown in figure 2. Over the years, researchers have developed mechanisms and tools for the automatic detection of types of learning style [43], [44], [45], [58], [64], [65], [66], [69]. However, few of the studies have discussed the mechanisms for generating an adaptive course based on detected leaning styles based on learning objects and material already provided by teachers [67], [68], [70].

As a motivating example, let us assume two students with different profiles (learning styles) have the same rating on the same learning objects. Clearly, from a fully personalised perspective, the $Top\_N$ LOs list based on predicted rating cannot be the same for both students. Hence, it is important to consider students learning styles when predicting their rating so as to improve the recommendation process. Accordingly, in this study, a novel hybrid recommendation algorithm is proposed based on the personalised students profile presented in [69] and K-means Clustering as a way to overcome information overload and cold start problems, thus building an effective course learning objects recommendation system.

### 1.1 Literature gap and research contributions

Recently published relevant research papers in the field of the e-learning recommendation systems, including the content-based filtering, collaborative filtering and hybrid recommendation techniques, are presented in Table 1. The current e-learning recommendation systems face the following problems.

1) First, the majority of the traditional recommendation algorithms have been developed for e-commerce applications that are unable to cover all the requirements of learning environments. In particular, they do not consider the learning process in their recommendation approach.
2) Second, the recommendation mechanisms that rely exclusively on two dimensions (i.e. users and items) dont consider the attributes of learners and learning materials [16]. As a result, rich and vital information, such as learners learning styles and the properties of learning objects are overlooked.
3) Third, during the continuous learning process, learners do not actively make ratings or give comments, because they aim to achieve their goals within scheduled but limited learning time. As a result, learners learning profiles often seem isolated from each other. The extreme data sparsity caused by these factors can render traditional recommendation techniques ineffective.
4) Finally, traditional recommendation methods have a low ability to capture and perceive the changes in learners preferences in an adaptive way [72].

In order to avoid these drawbacks and improve the accuracy of Top-N course learning objects recommendation. Our key contributions follow below:

1) In comparison with most of existing e-learning recommendation systems such as [39], [86], [89]

TABLE 1: Summary of existing personalised e-leaning recommendation systems

| | | | | Used Recommendation Techniques | | |
|---|---|---|---|---|---|---|
| **Study** | **Description** | **Cold-Start** | **Issue** | **CF** | **CBF** | **HF** |
| [89] | Predict the most suitable learning materials to each learner based on collaboration with other learners. | ✓ | No past preferences | ✓ | | |
| [39] | Recommend the most suitable learning materials for each learner based on the rating similarity with other learners. | ✓ | No past preferences or No high rated LO in past preferences | ✓ | | |
| [38] | Finding learning objects that would be suitable for learners preferences (knowledge level and learning style). | ✓ | No past preferences | | ✓ | |
| [37] | Recommend course learning objects based on neighborhood rating. | ✓ | No high rated LO in past preferences | ✓ | | |
| [36] | Constructed a course ontology and retrieved the course according to in learners learning styles. | ✓ | No past preferences | ✓ | | ✓(+ontology) |
| [91] | Proposed a hybrid recommender system for learning materials by combining CBF, CF and ontology. | ✓ | No past preferences or No high rated LO in past preferences | ✓ | ✓ | ✓(+ontology) |
| [46] | Suggested a method of clustering learning objects to improve their recommendations. | ✓ | Zero rating to the LO | ✓ | | ✓(+ k-means) |
| [91] | Proposed a hybrid recommender system to recommend learning items in users learning processes | ✓ | No past preferences or No high rated LO in past preferences | ✓ | | ✓(+SPM algorithm) |
| [73] | Proposed a recommender system for storing and sharing research papers and glossary terms among university students and industry practitioners. | ✓ | No past preferences | ✓ | | |
| [92] | Proposed the courseware management architecture with courseware recommendation that combines the user contents filtering and collaborative filtering. | ✓ | No past preferences | ✓ | ✓ | |
| [75] | proposed course recommendation system based on ontology and context aware e-learning. | ✓ | No high rated LO in past preferences | | ✓ | |
| [76] | Recommend learning contents to users based on similarity between user profiles. | ✓ | No past preferences | ✓ | | |
| [77] | Proposed framework for recommending learning materials based on the similarity of content items and good learners average rating strategy. | ✓ | No past preferences or No high rated LO in past preferences | | ✓ | |
| [80] | proposed a framework for recommending learning resources based on the learners recent navigation history and by comparing similarities and differences among different learners preferences and instructional content available in the e-learning system. | ✓ | No past preferences or No high rated LO in past preferences | ✓ | ✓ | |
| [81] | Recommend learning Materials based on multidimensional attributes | ✓ | No high rated LO in past preferences | ✓ | ✓ | |
| [82] | Proposed approach for selecting and sequencing the most appropriate learning objects. | ✓ | No high rated LO in past preferences | ✓ | | (+association pattern analysis) |

employed only rating values, the proposed algorithm takes into account multidimensional-attribute of learning objects and students learning styles accompanied by rating values in its rating prediction process. Thus, compared to these methods, the proposed method is more accurate facing with the sparse data and cold-start (Section 3.1)).

2) A new approach is proposed to overcome the new-learner zero-rated profile recommendation issue by determining the nearest learners with a similar historical rating and similar learning styles profile (Section 3.1.3).

3) Several recommendations algorithms will be tested in order to find out which is the one that works better for the course learning objects recommendation. The accuracy of the recommendations is measured using traditional evaluation metrics, namely the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). The results indicate that the hybrid recommender technique has higher accuracy in comparison with collaborative and content based recommendation techniques(Section 4).

Our proposed algorithm has been implemented in C++ using Visual Studio and Windows Presentation Foundation (WPF) to design the Graphical User Interface (GUI). It has been evaluated using a real student dataset from AASTS MOODLE (Arab Academy for Science and Technology and Maritime Transport - Modular Object-Oriented Dynamic Learning Environment).

The remainder of the paper is organised as follows. The next section defines the main concepts used in the proposed approach. Section3 discusses the proposed recommender algorithm, whilst Section 4 presents the experimental results and analysis. Section 4 concludes the paper and future research directions are proposed.

## 2 PRELIMINARIES

In the following sub-sections, we present techniques commonly used in recommender systems. These will be analysed later to improve the accuracy of recommendations. An overview of the similarity metrics and K-means clustering algorithm is also given.

### 2.1 Recommendation techniques

The underlying techniques used in recommender systems can be categorised into two broad classes: (a) content-based recommendation (b) collaborative filtering recommendation. New hybrid recommendation algorithms can be generated by synthesising these two methods [21].

#### *2.1.1 Content-Based Filtering (CBF)*

As a traditional recommendation method, the rationale for CBF is simple. The items recommended by this method are similar to the items of users interest [87], with matching information between items and users being the key procedure. In e-learning recommender systems, the items are the learning objects in the e-learning systems and the users are the learners. CBF recommender systems work with profiles of learners that are created at the beginning. A profile has information about a learner and his/her preferences, which are based on how he/she rates the LOs. Generally, when creating a profile, recommender systems make a survey to get initial information about a user in order to avoid the new-user problem. In the recommendation process, the engine compares the LOs that have already been positively rated by learner with LOs he/she has not done so and looks for similarities. Those LOs that are mostly similar to the positively rated ones, will be recommended to the user. In this case, the profiles of other users are not essential and they do not influence the recommendations of the user, for they are based on individual information.Figure 3 presents an illustrative example of CBF. From the figure 3, we can see the recommendation progress according to three main steps: Item Representation, Profile Learning and Recommendation Generation. As an example for e-learning application, [80] used learners recent navigation histories and similarities and dissimilarities among the contents of the learning materials for online automatic recommendations. Clustering was proposed by [88] to group learning documents based on their topics and similarities. Since in the e-learning environment learning materials are in a variety of multimedia formats, including text, hypertext, image, video, audio and slides, it is difficult to calculate the content similarity of two items [91]. In fact, the existing metrics in CBF only detect similarity between LOs that share the same attributes. This causes overspecialised recommendations that only include LOs very similar to those that the learner already knows.

#### *2.1.2 Collaborative Filtering (CF)*

Collaborative filtering became one of the most researched techniques of recommender systems after it was proposed and described by [60]. CF [17] recommends to the target

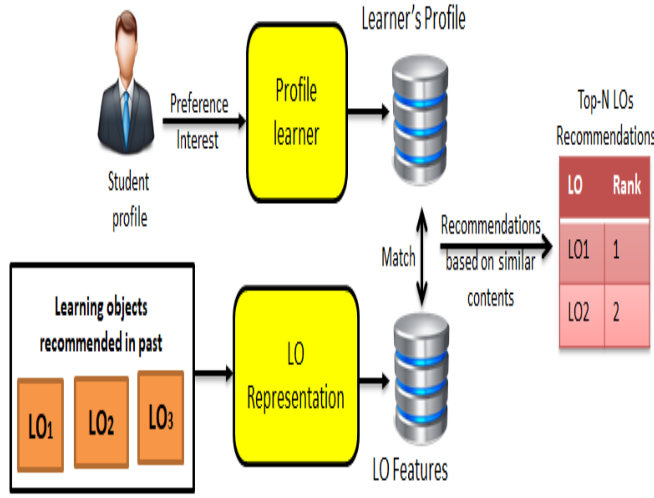



Fig. 3: Content-based filtering recommendation

learner learning resources that other similar learners have registered as liking previously. In other words, an important step in CF is to identify those learners most similar to the target learner. The similarity in taste of two learners is calculated based on their rating history. If two learners have the same or almost the same rated LOs in common, then they are deemed to have similar tastes. Such learners and others of the same ilk comprise a group or a so-called neighbourhood. A learner gets recommendations to choose LOs that he/she has not rated before, but have already been positively rated by those in his/her neighbourhood, as shown in Fig. 4. To this end, several research efforts have been made

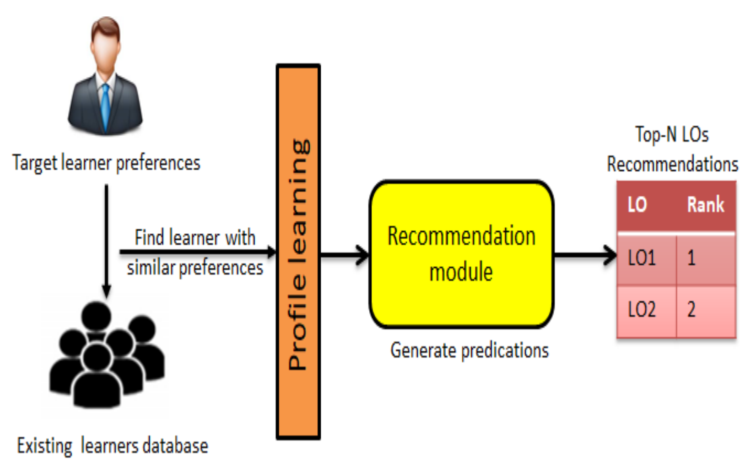



Fig. 4: Collaborative filtering algorithm

to identify similarity measures so as to identify these users with common profiles [84] [85]. CF was used by [89] for prediction of the most suitable materials for the learner as follows. First, the weight between all users and the active learner is calculated by Pearson correlation. Then, the n users that have the highest similarity to the active learner are selected as belonging to the neighborhood. Finally, using the weight combination obtained from the neighbourhood, the rating prediction is calculated. Regardless of its success in many application domains, collaborative filtering has two serious drawbacks. First, its applicability and quality are limited by the so-called sparsity problem, which occurs when the available data are insufficient for identifying similar users. Second, it requires knowing many user profiles in order to elaborate accurate recommendations for a given user. Given in some e-learning environments the learner population is low, recommendation results with this method in such cases will have poor accuracy.

#### 2.1.3 Hybrid Filtering (HF)

In the last few years, researchers of recommender systems have explored hybridisation of recommendation techniques as an approach for developing effective recommender systems. Table 1 lists some of the techniques that have been used to this end. Hybrid filtering entails combining two or more recommendation techniques to improve performance, as shown in Figure 5 In [92], a combination of content-

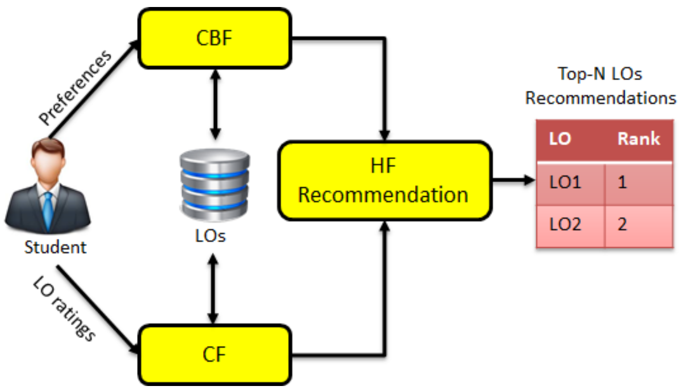



Fig. 5: Hybrid filtering algorithm

based and collaborative filtering was implemented to make personalised recommendations for a courseware selection module. The algorithm starts with user u entering some keywords on the portal of the courseware management system. Then, the courseware recommendation module finds within the same user interest group of user u the k courseware with the same or similar keywords that others have chosen. [23] applied association rule mining to identify interesting information through students usage data in the form of IF-THEN recommendation rules and then, used a collaborative recommender system to share and score the recommendation rules obtained by teachers with similar profiles as well as other experts in education.

### 2.2 Similarity metrics

Similarity metrics are the backbone of CF and CBF helping to predict the ratings of unrated items. Regarding which, in this study, the two most convenient similarity metrics, namely, the Pearsons correlation and cosine similarity [70] are considered.

#### 2.2.1 Pearson Correlation

The Pearson correlation coefficient is a measure of the linear dependence between two variables (real-valued vectors). Specifically, that of two variables $x$ and $y$ is formally defined as the covariance of the two variables divided by the product of their standard deviations (which acts as a normalisation factor) [78] and it can be equivalently defined by Eq. 1).

$$P(x,y) = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} \quad (1)$$

where $\overline{x}$ and $\overline{y}$ are the mean values of $x$ and $y$, respectively.

The coefficient $P(x,y)$ ranges from $-1$ to $1$ and is invariant to linear transformations of either variables. The value $-1$ represents perfect negative linear dependence, $0$ no linear dependence, and $1$ perfect positive linear dependence. Used as a similarity metric, negative values indicate dissimilarity, while positive values measure the similarity between the two variables with $1$ be the perfect similarity.

#### 2.2.2 Cosine Similarity

The cosine similarity involves measuring the angle between two vectors [25] and is calculated by Eq. (2), as the ratio of the scalar product by the product of the magnitudes.

$$c(x,y) = \frac{x.y}{||x||.||y||} \quad (2)$$

The values of $c(x,y)$ range from $-1$ to $1$ in general, and from $0$ to $1$ if the coordinates of $x$ and $y$ are non-negative values. This paper is interested in the latter where the value $0$ represents no similarity and $1$ perfect similarity.

### 2.3 K-means clustering algorithm

Clustering is one of the most common data mining techniques used in recommendation systems in order to develop recommendation rules or build recommendation models from large data sets [79]. It can be defined as the process of organising objects in a database into clusters (or groups), such that objects within the same cluster have a high degree of similarity, while those belonging to different ones have a high degree of dissimilarity. The K-means algorithm [71] is one of the most popular clustering algorithms due to its simplicity and intuitive interpretation. The algorithm has the following steps.

**Step 1:** Select K random points from the dataset as initial cluster centroids.

**Step 2:** Create K clusters by associating each data point with its closest cluster centroid, according to the Euclidean distance defined by Eq. (3).

$$D(x,y) = \sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \quad (3)$$

**Step 3:** Recalculate the centroid of each cluster as the mean of all the data points in that cluster.

**Step 4:** Repeat steps 2 and 3 until the centroids no longer change.

In our proposed system, we apply data clustering by employing the K-means algorithm to improve computational efficiency in terms of accuracy and quality of recommendations. To begin with, k initial cluster centres are identified and then the given data set is iteratively refined.

### 2.4 Description of the felder-silverman learning style model

The term 'learning styles' refers to the preferential way in which the student perceives, processes, understands and retains information [50]. Various learning style models have been presented in the past by researchers, such as those of Felder and Silverman [19], Honey and Mumford [51], Kolb citekolb1984englewood, Mayer and Myers [62], Dunn [61] and Pask [53]. In our case, we use the Felder and Silverman model (FSLSM) [19] to represent both the student learning styles and the learning object profiles for the following reasons.

- First, it is the most widely used in educational systems thanks to its ability to quantify students learning styles, as shown in 2.
- Second, it is used very often in technology enhanced learning and some researchers even argue that it is the most appropriate learning style model for the use in adaptive learning systems such as [20], [21], [22], [23], as well as being easy to implement [33], [63].

FSLSM describes learning styles by characterising each learner according to four dimensions, each of which, is defined as below.

The *information processing* dimension (active/reflective) tells how one prefers to process information. An active learner wants to try things out, working with others in groups, whilst a reflective one chooses to think things through, working alone or with a familiar partner.

The *information input* dimension (visual/verbal) determines how ones prefers information to be presented. A visual learner likes visual presentations, pictures, diagrams, and flow charts. A verbal learner prefers written and spoken explanations.

The *information understanding* dimension (sequential/global) determines how ones prefers to organise and progress towards understanding information. A sequential learner prefers linear thinking and learning in small incremental steps. By contrast, a global learner prefers holistic thinking, systems thinking, and learns in large leaps.

The *information perception* dimension (sensing/intuitive) states how you prefer to perceive or take in information. A sensing learner is attracted to concrete thinking, is practical as well as being concerned about facts and procedures. While an intuitive learner opts for conceptual thinking, being innovative, as well as being interested in theories and meanings.

It should be noted that each of these dimensions is characterised by a pair X/Y of learning style attributes (i.e. active/reflective, sequential/global, visual/verbal, and sensing/intuitive) meaning that the learning style of a learner in a particular dimension ranges from perfect X to perfect Y. For example, in the information processing dimension, the learning style of a student can be 70% *active* and 30% *reflective*. Of course, the percentage of X and the percentage of Y must sum up to 100%. Felder and Silverman [19] developed an Index of Learning Styles (ILS) questionnaire, which comprises 44 questions that has been shown to be effective in identifying the learning style of each individual learner. ILS provides a method of calculating the percentage values of learning style attributes from the learners answers to the questionnaire [19], [69].

The next section presents a novel algorithm for recommending learning objects based on student learning style.

## 3 Proposed algorithm for recommending personalised learning objects

In this work, a new algorithm for rating prediction of the learning objects is proposed. The proposed algorithm predicts the ratings for a given LO, for a given student based on learning styles, and their rating. The FSLSM learning style model described above is adopted to represent both the student learning preferences and the learning object profiles. The rating is given on a scale of 1-5. First, work is carried out to design an effective algorithm for recommending Top-N personalised learning objects in e-Learning systems based on student learning styles, as presented in (Sect. 3.1). Then an experimental study undertaken to find out which algorithm produces the best accuracy for rating prediction. The best performing algorithm is then retained for the recommender system. To present this study clearly, we first provide two definitions.

**Definition 1 (Student Profile).** It is assumed that the student learning style is represented by a vector of real values ranging from 0 to 1 (or from 0% to 100%) as follows, where the prefixes of learning style attributes are used as place holders.

$$LS = (act, ref, vis, ver, seq, glo, sen, int) \tag{4}$$

Some examples of student learning style vectors are given in Table 3 and these can be calculated using the learners responses to the ILS questionnaire [69] or according to his/her learning behaviour [70].

**Definition 2 (Object Profile).** The learning content materials are structured into learning objects for each topic. Learning objects are provided in various formats and media in order to meet the learning styles of individual learners. They can be text documents (e.g. pdf), presentations (e.g. powerpoint slides), images, audios, videos, simulations, etc. For example, a visual learner will prefer to watch a video than to read a pdf document, while a verbal one will choose to do opposite. Hence, a learning object profile (OP) can be represented by a FSLSM learning style vector indicating the category of learners that this learning object is suitable for, as in Eq.(5).

$$OP = (act, ref, vis, ver, seq, glo, sen, int) \tag{5}$$

Unlike the student learning styles that are calculated through the ILS questionnaire or behaviour, it is assumed that the learning object profile is set by the teacher or an education professional. Some examples of learning object profiles are given in Table 4 for illustration.

In the remaining part of the subsection, we give a description of the proposed predicated rating algorithms in detail and analyse the accuracy of the recommendations. Finally, we present the prediction algorithm proposed in our work.

TABLE 2: Summary of existing personalised e-leaning systems considering learning styles

| System Name | LS Model | Adaptation technique |
|---|---|---|
| Protus 2.0 [95] | FSLSM | Learning style identification & personalised content recommendation |
| WELSA [54] | Unified LS Model | Presentation recommendation & sequencing |
| MAS-PLANG [55] | FSLSM | Content personalisation |
| UZWEBMAT [96] | VARK | Content recommendation |
| PLORS [56] | FSLSM | learning objects recommendation |
| Tortorella and Graf [94] | FSLSM | Video, audio, presentation score calculation in mobile environment |
| Christudas [97] | FSLSM | Compatibility and complexity of learning objects |
| e-Teacher [98] | FSLSM | Course personalisation |
| OSCAR CITS [99] | FSLSM | Course personalisation ( SQL tutorial) |
| TANGOW [57] | FSLSM | Adapt course structure & sequencing |
| iWeaver[18] | Dunn & Dunn Model | Media recommendation (Flash animations or streaming audio) |
| learnFit [58] | MBTI | Course personalisation (PHP material ) |
| CS388 [58] | FSLSM | Course personalisation |
| Kurilovas [93] | Honey & Mumford learning style | Teaching/learning strategies |

TABLE 3: Examples of student learning style vectors

| | act | ref | vis | ver | seq | glo | sen | int |
|---|---|---|---|---|---|---|---|---|
| **Fatima** | 0.7 | 0.3 | 0.2 | 0.8 | 0.5 | 0.5 | 0.6 | 0.4 |
| **Tom** | 0.4 | 0.6 | 0.1 | 0.9 | 0.7 | 0.3 | 0.8 | 0.2 |
| **Clara** | 0.5 | 0.5 | 0.6 | 0.4 | 0.8 | 0.2 | 0.7 | 0.3 |

TABLE 4: Examples of learning object profile

| | act | ref | vis | ver | seq | glo | sen | int |
|---|---|---|---|---|---|---|---|---|
| $\mathbf{OP_1}$ | 0.7 | 0.3 | 0.5 | 0.5 | 1 | 0 | 0.3 | 0.7 |
| $\mathbf{OP_2}$ | 0.2 | 0.8 | 0.8 | 0.2 | 1 | 0 | 0.4 | 0.6 |
| $\mathbf{OP_3}$ | 0.5 | 0.5 | 0.2 | 0.8 | 0.1 | 0.9 | 0.6 | 0.4 |
| $\mathbf{OP_4}$ | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.3 | 0.7 |
| $\mathbf{OP_5}$ | 0.9 | 0.1 | 0.7 | 0.3 | 0.3 | 0.7 | 0.1 | 0.9 |

### 3.1 Rating prediction algorithms

In this step, the algorithm predicts the rating for a given learner for a given set of learning objects using prediction approaches. The following sections will briefly explain these approaches.

#### *3.1.1 Predicting ratings based on Collaborative Filtering*

From the description of the previous section, we notice that the traditional CF methods heavily rely on the co-rated items. However, the similarity computation cannot be performed when there are no rated items, which is called cold start problem (see Sect.1). For improving the accuracy and quality of recommendation, our research CF is implemented as follows:
Let $LS$ be the learning style vector of the active student.

1) Apply K-means to cluster the students profiles.
2) Select cs the nearest SP cluster to LS as in Eq. (1)
3) Foreach LO $x$
    a) Let $I =$ set of the top-n nearest elements to LS in cs **that have rated** $x$ as in Eq.(2)
    b) If $\|I\| > 0$ then calculate the predicted rating for $x$ as in Eq. (6)
    c) If $\|I\| = 0$ then calculate the predicted rating for $x$ as in Eq. (7)
4) Recommend the top-n highly rated LOs.

$$\tilde{r}_1(LS,x) = \frac{\sum_{u\in I} sim(LS,u) \times r(u,x)}{\sum_{u\in I} sim(LS,u)} \tag{6}$$

Where $\tilde{r}(LS,x)$depicts the predicted value of $LO_x$ of the active student $LS$. $sim(LS,u)$ donates to the similarity between student LS and other students who rated the $LO_x$.$\sum_{u\in I} sim(LS,u)$denotes the total similarities of students who rated the $LO_x$.

$$\tilde{r}_2(LS,x) = int(0.5 + sim(LS,x) \times 5) \tag{7}$$

The solution of cold-start (case $\|I\| = 0$ ) proposed by measuring the similarity between the context of learning objects $LO_x$ and student profile to predict and fill in missing user ratings and subsequently improve the accuracy of collaborative filtering as Eq.( 7).

#### *3.1.2 Predicting ratings based on Content-based filtering*

As explained in section 2.1,the general principle of content-based approaches is to identify the common characteristics of learning objects that have received a favorable rating from a learner, and then recommend to him/her new LO that share these characteristics. In this work, we proposed an algorithm of a similarity model to enhance the accuracy or recommendation, the similarity between $SP$ and $LO$ is calculated using the following steps:
Let $LS$ be the learning style vector of the active student.

1) Let $\mathcal{O}$ be the set of all learning objects rated by $LS$.
2) If $\mathcal{O} \neq \emptyset$ then
    a) Apply K-means to cluster $\mathcal{O}$
    b) Foreach LO $x$
        i) Let $co_x =$ the nearest LO cluster to $x$ as in Eq. (1)
        ii) Let $J =$ set of the top-n nearest elements to $x$ in $co_x$as in Eq.(2)
        iii) Calculate the predicted rating for $x$ as in Eq. (8)
    c) Recommend the top-n highly rated LOs.
3) If $\mathcal{O} = \emptyset$ then
    a) Apply K-means to cluster all the learning learning objects
    b) Let $co =$ the nearest LO cluster to $LS$
    c) Foreach $x \in co$
        i) Calculate the predicted rating for $x$ as in Eq. (7)

d) Recommend the top-n highly rated LOs in $co$.

$$\tilde{r}_3(LS, x) = \frac{\sum_{u \in J} sim(x, u) \times r(LS, u)}{\sum_{u \in J} sim(x, u)} \tag{8}$$

#### 3.1.3 Predicting ratings based on Hybrid Filtering

For improving the accuracy and quality of recommendation, our research HF is implemented as follows:
Let $LS$ be the learning style vector of the active student.

1) Let $\alpha$ be the weight of CF in the hybrid model; $0 \leq \alpha \leq 1$.
2) Apply K-means to cluster the students profiles
3) Select cs the nearest SP cluster to LS
4) Let $\mathcal{O}$ be the set of all learning objects rated by $LS$.
5) Apply K-means to cluster $\mathcal{O}$
6) Foreach LO $x$
   a) Let $I$ = set of the top-n nearest elements to LS in cs **that have rated** $x$
   b) Let $co_x$ = the nearest LO cluster to $x$
   c) Let $J$ = set of the top-n nearest elements to $x$ in $co_x$
   d) If $\|I\| > 0$ and $\|J\| > 0$ then calculate the predicted rating for $x$ as in Eq. (9)
   e) If $\|I\| = 0$ and $\|J\| > 0$ then calculate the predicted rating for $x$ as in Eq. (8)
   f) If $\|I\| > 0$ and $\|J\| = 0$ then calculate the predicted rating for $x$ as in Eq. (6)
   g) If $\|I\| = 0$ and $\|J\| = 0$ then calculate the predicted rating for $x$ as in Eq. (7)
7) Recommend the top-n highly rated LOs.

$$\tilde{r}(LS, x) = \alpha \times \tilde{r}_1(LS, x) + (1 - \alpha) \times \tilde{r}_3(LS, x) \tag{9}$$

Note the in Eq. (9), the value of $\alpha$ is between 0 and 1. Here are some examples:

- $\tilde{r}(LS, x) = 0.5 \times \tilde{r}_1(LS, x) + (1 - 0.5) \times \tilde{r}_3(LS, x)$
  $= 0.5 \times \tilde{r}_1(LS, x) + 0.5 \times \tilde{r}_3(LS, x)$
  $= \frac{\tilde{r}_1(LS,x) + \tilde{r}_3(LS,x)}{2}$
- $\tilde{r}(LS, x) = 0.2 \times \tilde{r}_1(LS, x) + (1 - 0.2) \times \tilde{r}_3(LS, x)$
  $= 0.2 \times \tilde{r}_1(LS, x) + 0.8 \times \tilde{r}_3(LS, x)$
- $\tilde{r}(LS, x) = 0.8 \times \tilde{r}_1(LS, x) + (1 - 0.8) \times \tilde{r}_3(LS, x)$
  $= 0.8 \times \tilde{r}_1(LS, x) + 0.2 \times \tilde{r}_3(LS, x)$
- $\tilde{r}(LS, x) = 0.75 \times \tilde{r}_1(LS, x) + (1 - 0.75) \times \tilde{r}_3(LS, x)$
  $= 0.75 \times \tilde{r}_1(LS, x) + 0.25 \times \tilde{r}_3(LS, x)$

"What is the recommendation algorithm that provides the best prediction of student ratings of learning objects?" An experimental study is carried out aimed at finding the answer to this question.

## 4 Experimental Study

The proposed recommendation system will suggest the most relevant learning objects to its learners from a large list. An experimental study was carried out to determine the most effective recommendation techniques to be used for the recommendation of LOs in e-learning systems. In this section, we describe the dataset 4.1,performance measurement 4.2 and the results 4.3 of our proposed approach with the existing approach.

### 4.1 Dataset

The dataset of the MOODLE log-file at AAST is used in this study for the autumn and spring semesters in 2016 and 2017 and 2017 and 2018 in the school of business. MOODLE (Modular Object-Oriented Dynamic Learning Environment) is defined as a course management system (CMS), being a free and open source software package designed using pedagogical principles, to help educators by creating effective online learning communities. The course of interest is on networks and e-commerce and comprises 20 topics, with each topic having multiple leaning objects in various presentation styles. There was a minimum of 15 learning objects for each topic. The experimental set up consisted of 80 students whose learning styles were identified using the ILS questionnaire, as explained in [69]. During the course, the students were asked to rate each learning object using a 5-level Likert scale, with 1 being not at all useful and 5 be very useful to their learning. In order to evaluate the different aspects of the proposed method, student dataset was split into different parts, including:

1) Cold-start students: a set of students with lower than 5 ratings;
2) Cold-start learning objects: a set of new LOs;
3) All students.

Cold-start was utilised to assess the ability of the algorithms to predict the ratings for those students a few LOs, so little information was available for these users. The goal was to investigate how additional sources of information, such as learning styles of a student, can be used along with rating information to improve the accuracy of rating prediction.

### 4.2 Performance measurement

In this analysis, accuracy metrics are considered to assess the performance of the proposed recommender system algorithm. We use statistical accuracy metrics to evaluate the accuracy of the rating prediction algorithm.
The frequently used statistical metrics are Mean Absolute Error (MAE) and Root-Mean-Square Error (RMSE). $r_i$ is the actual student rating of the learning object $i$ and $\tilde{r}_i$ is the predicted student rating for that learning object, $1 \leq i \leq n$.

In the computation of MAE, the first absolute sum of the difference between the actual and predicted rating is calculated, and then, it is divided by the total number of predicted ratings. Hence, a smaller value of MAE indicates a better accuracy of prediction, as in Eq. (10).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |r_i - \tilde{r}_i| \tag{10}$$

The Root Mean Square Error (RMSE) is calculated by dividing the sum of squares of the differences of the actual and predicted ratings by the total number of ratings on which the predictions are made. The RMSE is obtained by taking the square root of the MSE, as in Eq. (11).

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i - \tilde{r}_i)^2} \tag{11}$$

### 4.3 Experiments and evaluation

In our proposed work, a new approach to system for ratings prediction of the course learning objects is proposed. The prototype was implemented in C++ using Visual Studio and Windows Presentation Foundation (WPF) to design the GUI (graphical user interface), with the SQL server being utilised to allocate a system dataset and learner rating. A set of experiments was conducted on a Windows based PC with an Intel core i5 processor having a speed of 2.40 GHz and 16GB of RAM. The GUI, which allows for selecting various combinations of similarity metrics, is depicted in Fig. 6.

The first experiment was focused on the accuracy of rating predication, whereas the second one focused on the cold-start. Finally, the last part of the evaluation is about the integration of recommendation algorithms into AAST-MOODLE for testing them on real students.
Each experiment will be detailed and discussed in the following subsections.

#### *4.3.1 Evaluation on Rating Prediction*

After performing the preprocess on student dataset, 15 students were chosen randomly and the calculated predicted rating for these LO sets is shown in Table 5. The experimental results in the table show that the HF-0.5 algorithm has the best accuracy. From figure 7, it can be inferred that HF-0.5 has the least value of MAE and thus, provides better predictions. The MAE value of HF-0.5 is 0.9, whilst that of CBF is 1.52, which is greatest among all the three approaches. Hence, the latter method will produce the least accuracy in prediction. Theoretically and experimentally, it has already been proven that the root-mean-square error is always greater than the Mean Absolute Error.

Fig. 7 shows that the proposed HF-0.5 algorithm again delivers a smaller RMSE than other, which indicates that it is more accurate.

#### *4.3.2 Evaluation of Cold-start*

From another point of view, the experiments were repeated to evaluate the proposed approaches to handle cold-start problem recommendations for new students and new LOs.

- **New students:**The three different algorithms (CF, CBF and HF-0.5) can deal with new students by incorporating their personalised learning styles with their rating. Fig. 8a and Fig. 8b compare the accuracy of the different recommendation algorithms. The results achieved by the hybrid filtering approach are impressive. Given the above results, analysis, and discussion, it is concluded that the proposed algorithm HF-0.5 performs better than CF and CBF.
- **New learning objects:**Three algorithms can make recommendation for new LOs by measuring the similarity between learning object profile and student learning styles. From 9a and 9b we observe that HF-0.5 consistently outperforms in all the experiments, which indicates that our model handles new items better than CF and CBF.
- **New students and learning objects:** One special case is where neither the student nor the LOs exist in the previous user-item rating matrix. Most of the existing algorithms cannot deal with this situation. However, our proposed algorithm can still make recommendations by considering the relations between student and LO profiles.

The results achieved by the hybrid filtering approach (HF-0.5) are impressive. Given the above results, analysis, and discussion, it is concluded that the proposed algorithm performs better than CF and CBF.

#### *4.3.3 Real Student Evaluation*

The last part of the evaluation was to validate our method on real circumstances by integrating it in AAST MOODLE. The system was modified to be able to read student profiles and, subsequently, recommend the course LOs. To evaluate the student satisfaction with recommendations, a closed-ended questionnaire was administered to the 80 students who participated in the experiment. Previous studies on recommender systems have identified user satisfaction as one of the important evaluation measures [102], [103]. First, they were asked to fill in the FSLSM questionnaire [69] to create their profile, as shown in Fig. 10. After that, recommended five lessons for them over five weeks, with each including a set of LOs. We presented our empirical study to the Business Information Systems department at AAST. The questionnaire sought to find out whether the learner was satisfied or not satisfied with the LO recommendations. Fig. 11 illustrates the responses of the learners on satisfaction with these from each of the three recommendation algorithms.

From the figure 11, it is evident that the majority (95%) of the students were satisfied with the LO recommendations from the HF-0.5 algorithm. On the other hand, just (60% ) and (56%) of the students were satisfied with the recommendations from the CF and CBF recommendation algorithms, respectively.

## 5 Conclusion and Future Directions

Whilst recommender systems have been studied in the past decade, the study of rating predication for recommender systems is a more recent phenomenon. In this paper, we have concentrated on improving the accuracy and quality of recommendation in case of cold start and sparsity data. To this end, an improved rating predication algorithm has been proposed. To test the performed to compare the hybrid filtering, collaborative filtering, and content based filtering algorithms. The results of the implementation suggest that the HF-0.5 algorithm predicts better ratings to LOs as the value of the Mean Absolute Error is less than other algorithms. In addition to addressing both new students and new LOs issues in this context, the proposed recommender algorithm in the present research seems to improve the accuracy of recommended items to new students. However, this work

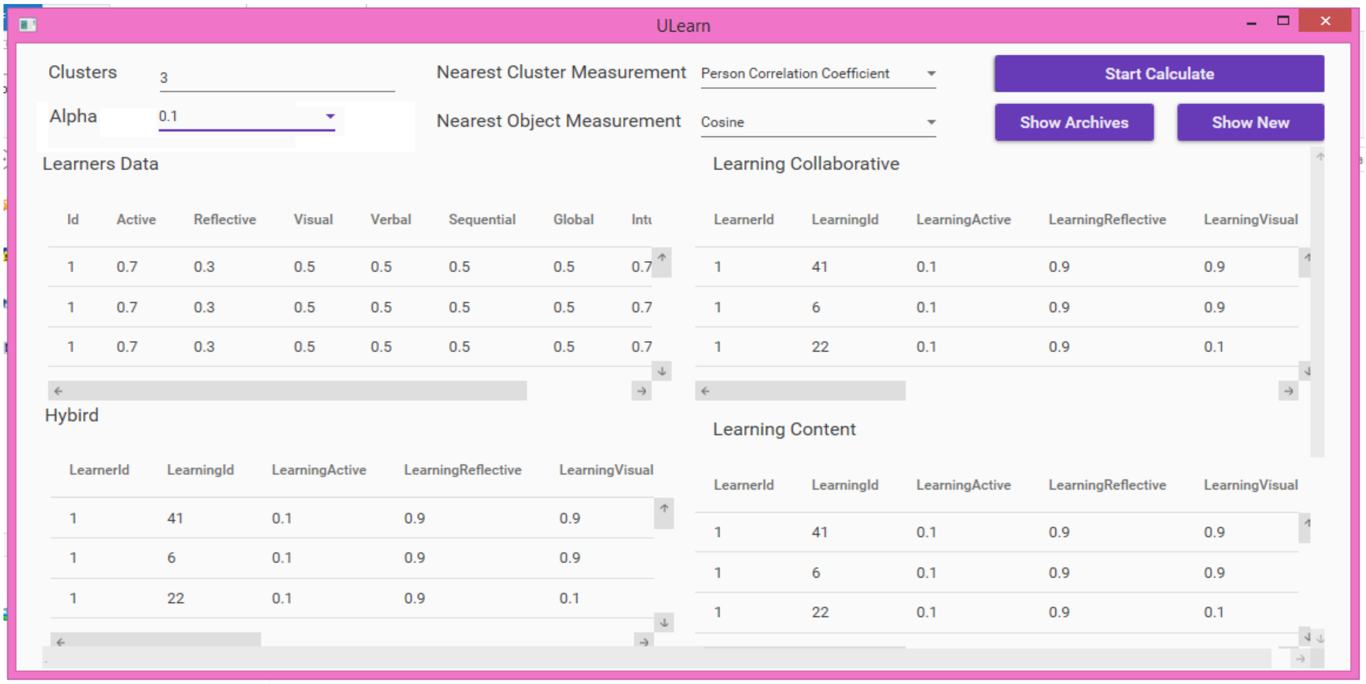



Fig. 6: ULearn interface

TABLE 5: Showing predicted rating using the proposed algorithm

| | | | Predicted rating using | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| S.ID | LO.ID | Act. R | CBF | CF | HF-0.1 | HF-0.2 | HF-0.3 | HF-0.4 | HF-0.5 | HF-0.6 | HF-0.7 | HF-0.8 | HF-0.9 |
| 1 | 1 | 2 | 1 | 2 | 1 | 2 | 2 | 2 | 2 | 3 | 3 | 2 | 1 |
| 2 | 5 | 4 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 4 |
| 3 | 13 | 2 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 2 | 3 | 3 | 3 |
| 4 | 25 | 5 | 2 | 3 | 2 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 3 |
| 5 | 10 | 2 | 3 | 3 | 3 | 2 | 1 | 3 | 2 | 3 | 2 | 2 | 3 |
| 6 | 1 | 3 | 1 | 2 | 2 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 2 |
| 7 | 5 | 2 | 1 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 8 | 13 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 2 |
| 9 | 15 | 4 | 2 | 2 | 2 | 3 | 3 | 4 | 4 | 4 | 4 | 3 | 2 |
| 10 | 20 | 5 | 3 | 4 | 3 | 4 | 3 | 4 | 5 | 4 | 4 | 4 | 3 |
| 11 | 22 | 2 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 3 |
| 12 | 4 | 3 | 1 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 13 | 8 | 5 | 3 | 4 | 3 | 4 | 4 | 4 | 5 | 4 | 4 | 3 | 3 |
| 14 | 7 | 4 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 3 | 3 |
| 15 | 16 | 5 | 2 | 4 | 2 | 3 | 5 | 5 | 4 | 5 | 4 | 4 | 3 |

Act. R= Actual Rating CBF= Content Based Filtering CF=Collaborative Filtering HF=Hybrid filtering
S.ID= Student ID LO.ID= Learning Object ID

has some limitations, which could be addressed in future work. First, the dataset in the current work was quite small and a larger one would add more weight to the findings. A second future direction is working on some other challenges of recommendation systems, such as scalability.

## References

[1] Sosnovsky, S., & Brusilovsky, P. (2015). Evaluation of topic-based adaptation and student modeling in QuizGuide. User Modeling and User-Adapted Interaction, 25(4), 371-424.

[2] A. Bennane, "Adaptive educational software by applying reinforcement learning." *Informatics in Education*, vol. 12, no. 1, 2013.

[3] K. I. Ghauth and N. A. Abdullah, "Learning materials recommendation using good learners ratings and content-based filtering," *Educational technology research and development*, vol. 58, no. 6, pp. 711–727, 2010.

[4] P. Vu, S. Fredrickson, and C. Moore, *Handbook of Research on Innovative Pedagogies and Technologies for Online Learning in Higher Education*. IGI Global, 2016.

[5] A. R. Anaya, M. Luque, and T. García-Saiz, "Recommender system in collaborative learning environment using an influence diagram," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7193–7202, 2013.

[6] I. Kamsa, R. Elouahbi, F. El Khoukhi, T. Karite, and H. Zouiten, "Optimizing collaborative learning path by ant's optimization technique in e-learning system," in *Information Technology Based Higher Education and Training (ITHET), 2016 15th International Conference on*. IEEE, 2016, pp. 1–5.

[7] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge & Data Engineering, (6), 734-749.

[8] Bourkoukou, O., & El Bachari, E. (2016). E-learning personalization based on collaborative filtering and learners preference. Journal of Engineering Science and Technology, 11(11), 1565-1581.

[9] Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In Recommender systems handbook (pp. 217-253). Springer, Boston, MA.

[10] Turnip, R., Nurjanah, D., & Kusumo, D. S. (2017, November). Hybrid recommender system for learning material using content-based filtering and collaborative filtering with good learners'

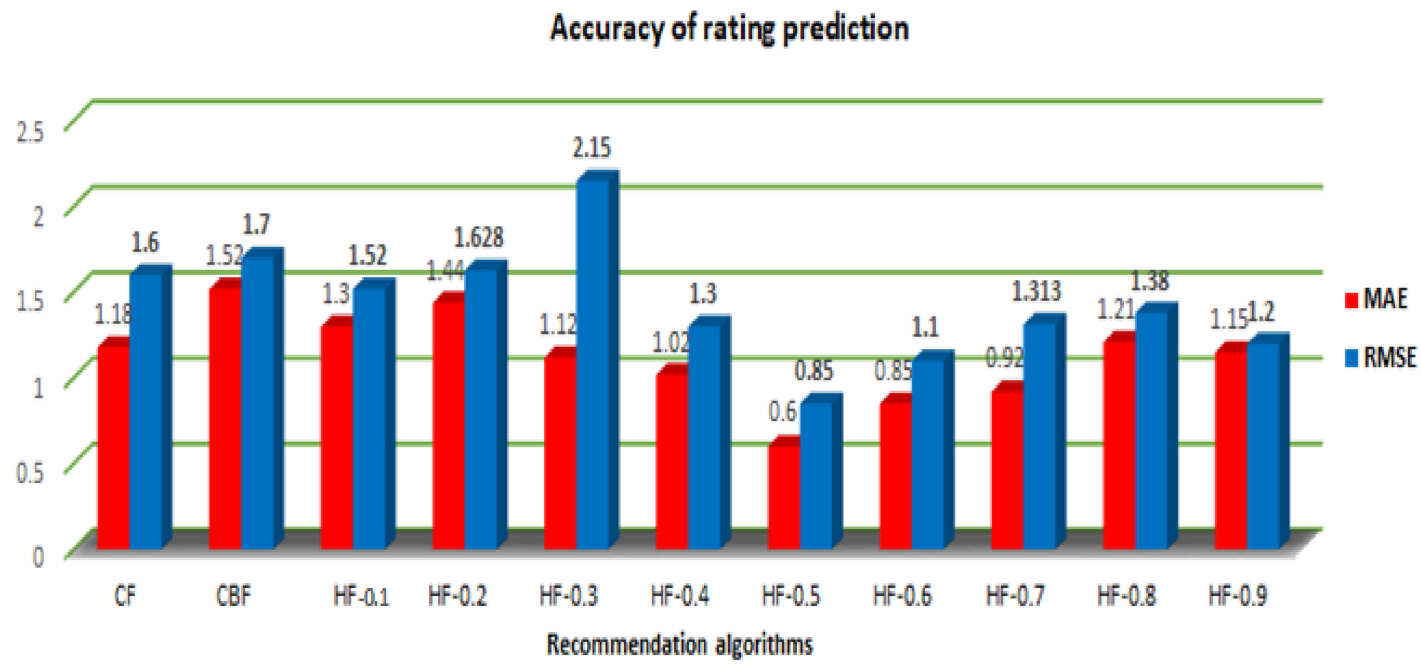
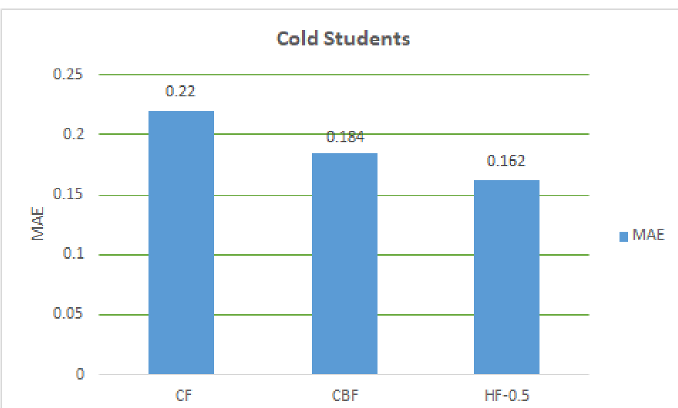
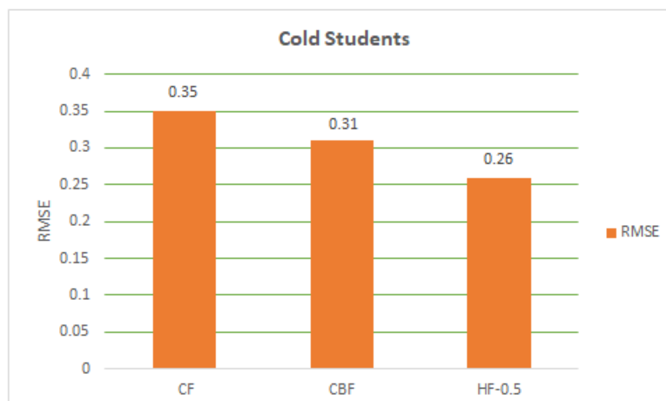


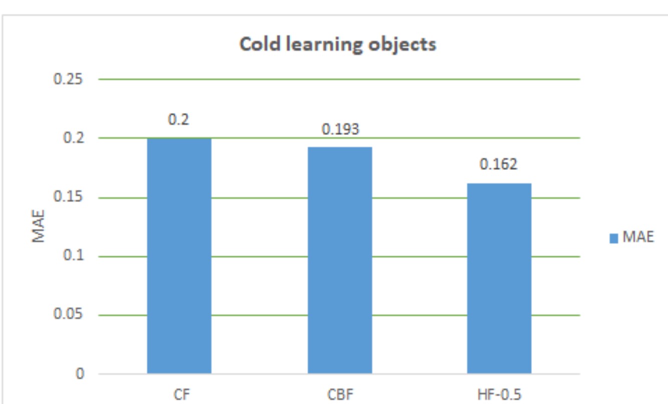

Fig. 7: Accuracy of the recommender algorithm using MAE and RMSE for m=50



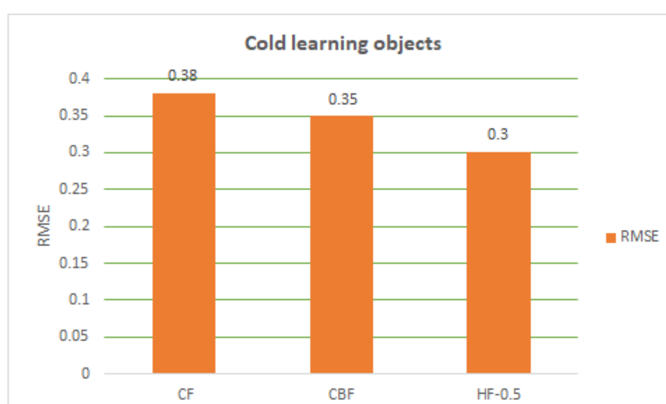

(a) Performance comparison for cold students using MAE




(b) Performance comparison for cold students using RMSE




(a) Performance comparison for cold LOs using MAE




(b) Performance comparison for cold LOs using RMSE

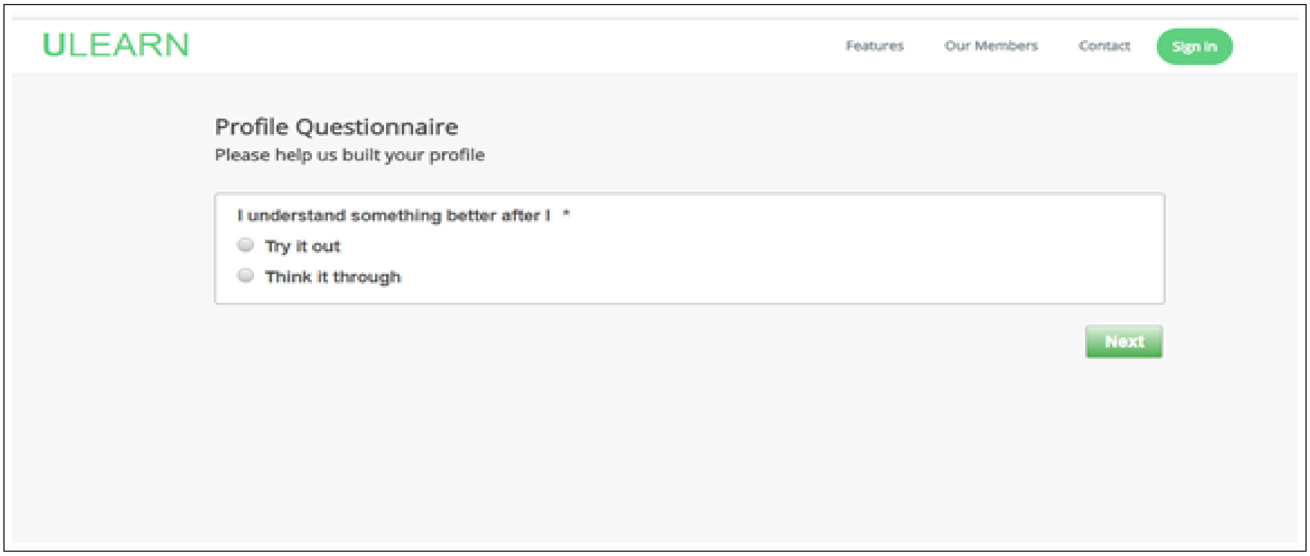
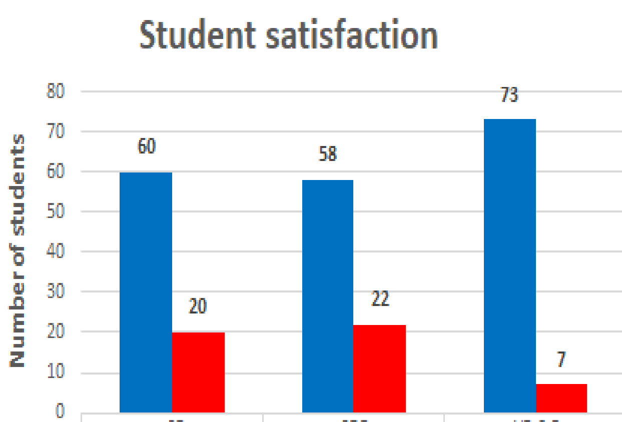



Fig. 10: Questionnaire interface




| | CF | CBF | HF-0.5 |
|---|---|---|---|
| satisfied | 60 | 58 | 73 |
| Not satisfied | 20 | 22 | 7 |

Fig. 11: Student satisfaction with LO recommendations

rating. In 2017 IEEE Conference on e-Learning, e-Management and e-Services (IC3e) (pp. 61-66). IEEE.
[11] Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In The adaptive web (pp. 291-324). Springer, Berlin, Heidelberg.
[12] Yao, L., Sheng, Q. Z., Ngu, A. H., Yu, J., & Segev, A. (2015). Unified collaborative and content-based web service recommendation. IEEE Transactions on Services Computing, 8(3), 453-466.
[13] Wei, J., He, J., Chen, K., Zhou, Y., & Tang, Z. (2017). Collaborative filtering and deep learning based recommendation system for cold start items. Expert Systems with Applications, 69, 29-39.
[14] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge & Data Engineering, (6), 734-749.
[15] da Silva, E. Q., Camilo-Junior, C. G., Pascoal, L. M. L., & Rosa, T. C. (2016). An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering. Expert Systems with Applications, 53, 204-218.
[16] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge & Data Engineering, (6), 734-749.
[17] Nt, J. A. R., Tomaz, L. F. C., De Souza, J. M., & Xexo, G. (2013). Bringing knowledge into recommender systems. Journal of systems and software, 86(7), 1751-1758.
[18] Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In The adaptive web (pp. 325-341). Springer, Berlin, Heidelberg.
[19] R. M. Felder, L. K. Silverman *et al.*, "Learning and teaching styles in engineering education," *Engineering education*, vol. 78, no. 7, pp. 674–681, 1988.
[20] C. A. Carver, R. A. Howard, and W. D. Lane, "Enhancing student learning through hypermedia courseware and incorporation of student learning styles," *IEEE transactions on Education*, vol. 42, no. 1, pp. 33–38, 1999.
[21] J. Kuljis and F. Liu, "A comparison of learning style theories on the suitability for elearning," *Web Technologies, Applications, and Services*, vol. 2005, pp. 191–197, 2005.
[22] F. Dağ and A. Geçer, "Relations between online learning and learning styles," *Procedia-Social and Behavioral Sciences*, vol. 1, no. 1, pp. 862–871, 2009.
[23] P. García, A. Amandi, S. Schiaffino, and M. Campo, "Evaluating bayesian networks precision for detecting students learning styles," *Computers and Education*, vol. 49, no. 3, pp. 794–808, 2007.
[24] R. M. Felder and J. Spurlin, "Applications, reliability and validity of the index of learning styles," *International journal of engineering education*, vol. 21, no. 1, pp. 103–112, 2005.
[25] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.
[26] Carr, S. (2000). As distance education comes of age, the challenge is keeping the students. Chronicle of higher education, 46(23).
[27] Inan, F. A., Yukselturk, E., & Grant, M. M. (2009). Profiling potential dropout students by individual characteristics in an online certificate program. International Journal of Instructional Media, 36(2), 163-177.
[28] Kotsiantis, S. B., Pierrakeas, C. J., & Pintelas, P. E. (2003, September). Preventing student dropout in distance learning using machine learning techniques. In International conference on knowledge-based and intelligent information and engineering systems (pp. 267-274). Springer, Berlin, Heidelberg.
[29] Willging, P. A., & Johnson, S. D. (2009). Factors that influence students' decision to dropout of online courses. Journal of Asynchronous Learning Networks, 13(3), 115-127.
[30] Niesler, A., & Wydmuch, G. (2009). User profiling in intelligent tutoring systems based on Myers-Briggs personality types. In Proceedings of the international multiconference of engineers and computer scientists (Vol. 1).
[31] Chen, C. M., Lee, H. M., & Chen, Y. H. (2005). Personalized e-learning system using item response theory. Computers & Education, 44(3), 237-255.
[32] Lu, J. (2004). A personalized e-learning material recommender

system. In International Conference on Information Technology and Applications. Macquarie Scientific Publishing.

[33] Lu, J. (2004). A personalized e-learning material recommender system. In International Conference on Information Technology and Applications. Macquarie Scientific Publishing.

[34] Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender systems handbook (pp. 1-35). Springer, Boston, MA.

[35] Soonthornphisaj, N., Rojsattarat, E., & Yim-Ngam, S. (2006, August). Smart e-learning using recommender system. In International conference on intelligent computing (pp. 518-523). Springer, Berlin, Heidelberg.

[36] Gulzar, Z., Leema, A. A., & Deepak, G. (2018). PCRS: Personalized course recommender system based on hybrid approach. Procedia Computer Science, 125, 518-524.

[37] Manouselis, N., & Costopoulou, C. (2007). Experimental analysis of design choices in multiattribute utility collaborative filtering. International Journal of Pattern Recognition and Artificial Intelligence, 21(02), 311-331.

[38] Yaghmaie, M., & Bahreininejad, A. (2011). A context-aware adaptive learning system using agents. Expert Systems with Applications, 38(4), 3280-3286.

[39] Recker, M. M., Walker, A., & Lawless, K. (2003). What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education. Instructional Science, 31(4-5), 299-316.

[40] Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet computing, (1), 76-80.

[41] Diao, Q., Qiu, M., Wu, C. Y., Smola, A. J., Jiang, J., & Wang, C. (2014, August). Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 193-202). ACM.

[42] El Bachari, E., Abelwahed, E. H., & El Adnani, M. (2012). An adaptive teaching strategy model in e-learning using learners preference: LearnFit framework. International Journal of Web Science 3, 1(3), 257-274.

[43] Klanja-Milievi, A., Vesin, B., Ivanovi, M., & Budimac, Z. (2011). E-Learning personalization based on hybrid recommendation strategy and learning style identification. Computers & Education, 56(3), 885-899.

[44] Kurilovas, E., Kubilinskiene, S., & Dagiene, V. (2014). Web 3.0Based personalisation of learning objects in virtual learning environments. Computers in Human Behavior, 30, 654-662.

[45] Latham, A., Crockett, K., & McLean, D. (2014). An adaptation algorithm for an intelligent natural language tutoring system. Computers & Education, 71, 97-110.

[46] Rodrguez, P., Heras, S., Palanca, J., Duque, N., & Julin, V. (2015). Argumentation-based hybrid recommender system for recommending learning objects. In Multi-Agent Systems and Agreement Technologies (pp. 234-248). Springer, Cham.

[47] dos Santos, H. L., Cechinel, C., Araujo, R. M., & Sicilia, M. . (2015, September). Clustering learning objects for improving their recommendation via collaborative filtering algorithms. In Research Conference on Metadata and Semantics Research (pp. 183-194). Springer, Cham.

[48] Winoto, P., & Tang, T. Y. (2010). The role of user mood in movie recommendations. Expert Systems with Applications, 37(8), 6086-6092.

[49] Nez-Valdz, E. R., Lovelle, J. M. C., Martnez, O. S., Garca-Daz, V., De Pablos, P. O., & Marn, C. E. M. (2012). Implicit feedback techniques on recommender systems applied to electronic books. Computers in Human Behavior, 28(4), 1186-1193.

[50] Pashler, H., McDaniel, M., Rohrer, D., & Bjork, R. (2008). Learning styles: Concepts and evidence. Psychological science in the public interest, 9(3), 105-119.

[51] Honey, P., & Mumford, A. (1982). The manual of learning styles (Maidenhead, McGraw-Hill).

[52] Kolb, D. A. (1984). Englewood Cliffs. New Jersey.

[53] Pask, G. (1976). Styles and strategies of learning. British journal of educational psychology, 46(2), 128-148.

[54] Popescu, Elvira, Costin Badica, and Lucian Moraret. ”Accommodating learning styles in an adaptive educational system.” Informatica 34.4 (2010).

[55] Pea, C. I., Marzo, J. L., & de la Rosa, J. L. (2002, September). Intelligent agents in a teaching and learning environment on the Web. In Proceedings of the international conference on advanced learning technologies (pp. 21-27). Palmerston North, NZ: IEEE Learning Technology Task Force.

[56] Imran, H., Belghis-Zadeh, M., Chang, T. W., & Graf, S. (2016). PLORS: a personalized learning object recommender system. Vietnam Journal of Computer Science, 3(1), 3-13.

[57] Paredes, P., & Rodriguez, P. (2004). A mixed approach to modelling learning styles in adaptive educational hypermedia. Advanced Technology for Learning, 1(4), 210-215.

[58] El Bachari, E., Abelwahed, E. H., & El Adnani, M. (2012). An adaptive teaching strategy model in e-learning using learners preference: LearnFit framework. International Journal of Web Science 3, 1(3), 257-274.

[59] Chen, W., Niu, Z., Zhao, X., & Li, Y. (2014). A hybrid recommendation algorithm adapted in e-learning environments. World Wide Web, 17(2), 271-284.

[60] Resnick, P., & Varian, H. R. (1997). Recommender systems. Communications of the ACM, 40(3), 56-59.

[61] Dunn, R. (1990). Understanding the Dunn and Dunn learning styles model and the need for individual diagnosis and prescription. Reading, Writing, and Learning Disabilities, 6(3), 223-247.

[62] Briggs-Myers, I., & Myers, P. B. (1995). Gifts differing: Understanding personality type.

[63] Hwang, G. J., Sung, H. Y., Hung, C. M., & Huang, I. (2013). A learning style perspective to investigate the necessity of developing adaptive learning systems. Journal of Educational Technology & Society, 16(2), 188-197.

[64] Chang, Y. C., Kao, W. Y., Chu, C. P., & Chiu, C. H. (2009). A learning style classification mechanism for e-learning. Computers & Education, 53(2), 273-285.

[65] zpolat, E., & Akar, G. B. (2009). Automatic detection of learning styles for an e-learning system. Computers & Education, 53(2), 355-367.

[66] Dora, F. A., Lima, L. V., Fernandes, M. A., & Lopes, C. R. (2013). Comparing strategies for modeling students learning styles through reinforcement learning in adaptive and intelligent educational systems: An experimental analysis. Expert Systems with Applications, 40(6), 2092-2101.

[67] Sangineto, E., Capuano, N., Gaeta, M., & Micarelli, A. (2008). Adaptive course generation through learning styles representation. Universal Access in the Information Society, 7(1-2), 1-23.

[68] Graf, S. (2009, September). Advanced adaptivity in learning management systems by considering learning styles. In Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 03 (pp. 235-238). IEEE Computer Society.

[69] Nafea, S. M., Siewe, F., & He, Y. (2018, September). ULEARN: Personalised Learners Profile Based On Dynamic Learning Style Questionnaire. In Proceedings of SAI Intelligent Systems Conference (pp. 1105-1124). Springer, Cham.

[70] Nafea, S. M., Siewe, F., & He, Y. (2019, February). A Novel Algorithm for Course Learning Object Recommendation Based on Student Learning Styles. In 2019 International Conference on Innovative Trends in Computer Engineering (ITCE) (pp. 192-201). IEEE.

[71] Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. IEEE Transactions on Pattern Analysis & Machine Intelligence, (7), 881-892.

[72] Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002, August). Methods and metrics for cold-start recommendations. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 253-260). ACM.

[73] Tang, T. Y., & McCalla, G. (2003, July). Smart recommendation for an evolving e-learning system. In Workshop on Technologies for Electronic Documents for Supporting Learning, International Conference on Artificial Intelligence in Education (pp. 699-710).

[74] Liang, G., Weining, K., & Junzhou, L. (2006, July). Courseware recommendation in e-learning system. In International Conference on Web-based Learning (pp. 10-24). Springer, Berlin, Heidelberg.

[75] Yu, Z., Nakamura, Y., Jang, S., Kajita, S., & Mase, K. (2007, July). Ontology-based semantic recommendation for context-aware e-learning. In International Conference on Ubiquitous Intelligence and Computing (pp. 898-907). Springer, Berlin, Heidelberg.

[76] Tan, H., Guo, J., & Li, Y. (2008, December). E-learning recommendation system. In 2008 International Conference on Computer Science and Software Engineering (Vol. 5, pp. 430-433). IEEE.

[77] Ghauth, K. I. B., & Abdullah, N. A. (2009, July). Building an e-learning recommender system using vector space model and good learners average rating. In 2009 Ninth IEEE International Conference on Advanced Learning Technologies (pp. 194-196). IEEE.

[78] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[79] Schafer, J. B. (2005). Application of Data Mining to Recommender Systems. In Encyclopedia of data warehousing and mining (pp. 44-48). IGI Global.

[80] Khribi, M. K., Jemni, M., & Nasraoui, O. (2008, July). Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. In 2008 Eighth IEEE International Conference on Advanced Learning Technologies (pp. 241-245). IEEE.

[81] Salehi, M., & Kmalabadi, I. N. (2012). A hybrid attributebased recommender system for elearning material recommendation. Ieri Procedia, 2, 565-570.

[82] Bourkoukou, O., El Bachari, E., & El Adnani, M. (2017). A recommender model in e-learning environment. Arabian Journal for Science and Engineering, 42(2), 607-617.

[83] Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In The adaptive web (pp. 291-324). Springer, Berlin, Heidelberg.

[84] Golbeck, J., & Hendler, J. (2006). Inferring binary trust relationships in web-based social networks. ACM Transactions on Internet Technology (TOIT), 6(4), 497-529.

[85] Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet computing, (1), 76-80.

[86] Ar, Y., & Bostanci, E. (2016). A genetic algorithm solution to the collaborative filtering problem. Expert Systems with Applications, 61, 122-128.

[87] Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In The adaptive web (pp. 325-341). Springer, Berlin, Heidelberg.

[88] Hammouda, K., & Kamel, M. (2006, April). Collaborative document clustering. In Proceedings of the 2006 SIAM International Conference on Data Mining (pp. 453-463). Society for Industrial and Applied Mathematics.

[89] Soonthornphisaj, N., Rojsattarat, E., & Yim-Ngam, S. (2006, August). Smart e-learning using recommender system. In International conference on intelligent computing (pp. 518-523). Springer, Berlin, Heidelberg.

[90] Bobadilla, J. E. S. U. S., Serradilla, F., & Hernando, A. (2009). Collaborative filtering adapted to recommender systems of e-learning. Knowledge-Based Systems, 22(4), 261-265.

[91] Chen, W., Niu, Z., Zhao, X., & Li, Y. (2014). A hybrid recommendation algorithm adapted in e-learning environments. World Wide Web, 17(2), 271-284.

[92] Liang, G., Weining, K., & Junzhou, L. (2006, July). Courseware recommendation in e-learning system. In International Conference on Web-based Learning (pp. 10-24). Springer, Berlin, Heidelberg.

[93] Kurilovas, E., Zilinskiene, I., & Dagiene, V. (2014). Recommending suitable learning scenarios according to learners preferences: An improved swarm based approach. Computers in Human Behavior, 30, 550-557.

[94] Tortorella, R. A., & Graf, S. (2017). Considering learning styles and context-awareness for mobile adaptive learning. Education and Information Technologies, 22(1), 297-315.

[95] Vesin, B., Ivanovi, M., KlaNja-MiliEvi, A., & Budimac, Z. (2012). Protus 2.0: Ontology-based semantic recommendation in programming tutoring system. Expert Systems with Applications, 39(15), 12229-12246.

[96] zyurt, ., zyurt, H., Baki, A., Gven, B., & Karal, H. (2012). Evaluation of an adaptive and intelligent educational hypermedia for enhanced individual learning of mathematics: A qualitative study. Expert Systems with Applications, 39(15), 12092-12104.

[97] Christudas, B. C. L., Kirubakaran, E., & Thangaiah, P. R. J. (2018). An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials. Telematics and Informatics, 35(3), 520-533.

[98] Schiaffino, Silvia, Patricio Garcia, and Analia Amandi. "eTeacher: Providing personalized assistance to e-learning students." Computers and Education 51.4 (2008): 1744-1754.

[99] Latham, A., Crockett, K., McLean, D., & Edmonds, B. (2012). A conversational intelligent tutoring system to automatically predict learning styles. Computers & Education, 59(1), 95-109.

[100] Guo, G. (2013, June). Improving the performance of recommender systems by alleviating the data sparsity and cold start problems. In Twenty-Third International Joint Conference on Artificial Intelligence.

[101] Brace, I. (2018). Questionnaire design: How to plan, structure and write survey material for effective market research. Kogan Page Publishers.

[102] Erdt, M., Fernandez, A., Rensing, C. (2015). Evaluating recommender systems for technology enhanced learning: a quantitative survey. IEEE Transactions on Learning Technologies, 8(4), 326-344.

[103] Jeremi, Z., Jovanovi, J., Gaevi, D., Hatala, M. (2009, September). Project-based collaborative learning environment with context-aware educational services. In European Conference on Technology Enhanced Learning (pp. 441-446). Springer, Berlin, Heidelberg.


**S** haimaa M. Nafea is a PhD student. She received her master's degree in information system from Arab academy for science and technology and maritime transport. Her main research interests are in adaptation and personalization of e-learning environment, learning styles, algorithms, recommender systems and collaborative filtering, machine learning.